Databanken – SQL – oefeningen

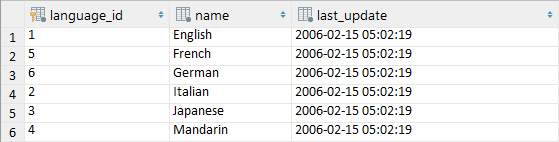
Maak gebruik van de **sakila** database om volgende oefeningen op te lossen.

1. Welke talen komen voor in de databank (alfabetisch gesorteerd)? Toon alle velden.

Antwoord:

SELECT \* FROM language

ORDER BY name asc;



1. Geef de volledige naam (voornaam en familienaam) van de acteurs/actrices met “SON” in hun familienaam, gesorteerd op voornaam.

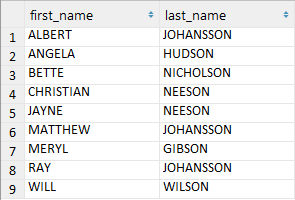
Antwoord:

SELECT first\_name, last\_name

FROM actor

WHERE last\_name LIKE '%SON%'

ORDER BY first\_name asc;



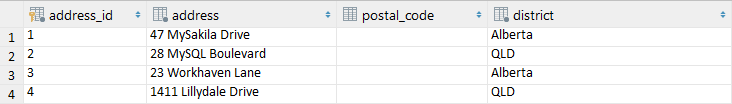
1. Geef een lijst van alle adressen zonder postcode. Toon de velden zoals hieronder.

Antwoord:

SELECT address\_id, address, postal\_code, district

FROM address

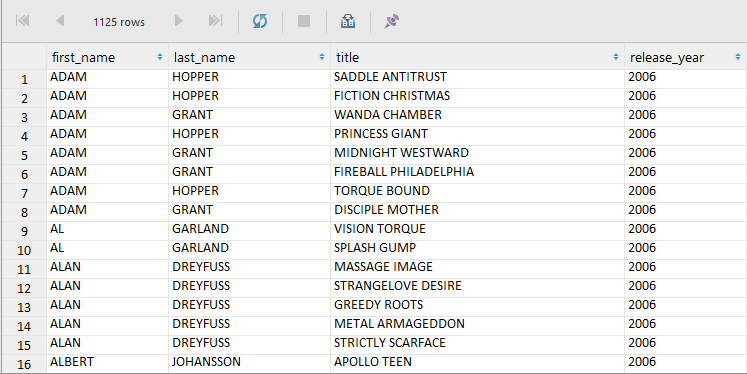
WHERE postal\_code = "";



1. Geef voor- en achternaam van acteurs/actrices die speelden in een film met “Crocodile” of “Shark” in de beschrijving, samen met de titel en het releasejaar van de film, gesorteerd op voornaam.

Antwoord:

SELECT first\_name, last\_name, title, release\_year  
FROM actor AS a  
 INNER JOIN film\_actor fa  
 ON a.actor\_id = fa.actor\_id  
 INNER JOIN film f  
 ON fa.film\_id = f.film\_id  
WHERE f.description LIKE '%shark%' OR f.description LIKE '%crocodile%'  
ORDER BY first\_name;



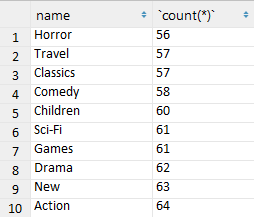
1. In hoeveel films komt een krokodil of een haai voor?

SELECT count(\*)  
FROM film  
WHERE film.description LIKE '%crocodile%'  
OR film.description LIKE '%shark%'  
OR film.title LIKE '%crocodile%'  
OR film.title LIKE '%shark%';

Antwoord: 203

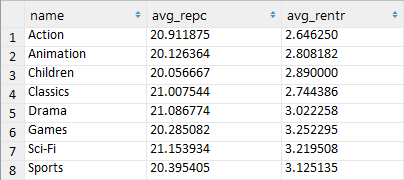
1. Zoek alle filmcategorieën waarin er tussen 55 en 65 films voorkomen. Geef de namen van deze categorieën, en het aantal films daarin, gesorteerd op het aantal films.

SELECT category.name, count(\*)  
FROM category  
 LEFT JOIN film\_category fc ON category.category\_id = fc.category\_id  
GROUP BY category.name  
HAVING count(\*) BETWEEN 55 AND 65  
ORDER BY count(\*);



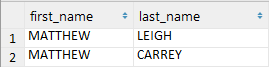
1. In hoeveel filmcategorieën is het verschil tussen de gemiddelde vervangkost (replacement cost) en de huurprijs (rental rate) groter dan 17? Geef een lijstje zoals hieronder.

SELECT category.name,  
 avg(film.replacement\_cost) AS avg\_repc,  
 avg(film.rental\_rate) AS avg\_rentr  
 FROM category  
 INNER JOIN film\_category  
 ON category.category\_id = film\_category.category\_id  
 INNER JOIN film  
 ON film\_category.film\_id = film.film\_id  
GROUP BY category.name  
HAVING (avg\_repc-avg\_rentr) > 17;



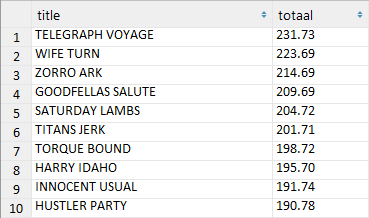
1. Geef de namen (voor- en achternaam) van alle acteurs en klanten wiens voornaam dezelfde is als de voornaam van de acteur met id 8. Neem de acteur met id 8 niet op in het lijstje. Je mag de naam van de acteur met id 8 niet gebruiken in je SQL statement, wel het id (het cijfer 8).

SELECT first\_name, last\_name  
FROM actor  
WHERE first\_name = (  
 SELECT first\_name  
 FROM actor  
 WHERE actor\_id = 8  
 )  
AND actor\_id !=8  
UNION #съединява два израза, за да покаже резултатите в една таблица  
SELECT first\_name, last\_name  
FROM customer  
WHERE first\_name = (  
 SELECT first\_name  
 FROM actor  
 WHERE actor\_id = 8  
 )



1. Geeft de top 10 van de best verhurende films (die het meest opbrachten volgens het totaal van betalingen in de tabel *payments*). Geef titel van de film, en totale opbrengst. Sorteer op totale opbrengst, aflopend.

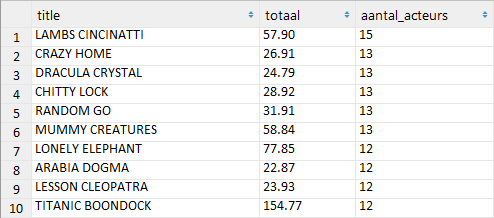
SELECT f.title, sum(p.amount) as totaal  
FROM film f  
 INNER JOIN inventory i on f.film\_id = i.film\_id  
 INNER JOIN rental r on i.inventory\_id = r.inventory\_id  
 INNER JOIN payment p on r.rental\_id = p.rental\_id  
GROUP BY f.title  
ORDER BY totaal desc  
LIMIT 10;



1. Geeft de top 10 van de films met de meeste acteurs. Geef titel van de film, aantal acteurs, en totale opbrengst. Sorteer op het aantal acteurs, aflopend.

SELECT f.title,  
 (SELECT sum(amount) FROM payment AS p  
 INNER JOIN rental r ON p.rental\_id = r.rental\_id  
 INNER JOIN inventory i ON r.inventory\_id = i.inventory\_id  
 INNER JOIN film f2 ON i.film\_id = f2.film\_id  
 WHERE f2.film\_id = f.film\_id ) AS totaal,  
COUNT(fa.actor\_id) AS aantal\_acteurs  
FROM film AS f

INNER JOIN film\_actor fa ON f.film\_id = fa.film\_id  
GROUP BY f.title  
ORDER BY aantal\_acteurs DESC  
LIMIT 10;



1. Geef de top 10 van landen met de meeste klanten.

SELECT country, count(c.customer\_id) AS aantal\_klanten  
FROM country  
 INNER JOIN city s ON country.country\_id = s.country\_id  
 INNER JOIN address a ON s.city\_id = a.city\_id  
 INNER JOIN customer c ON a.address\_id = c.address\_id  
GROUP BY country  
ORDER BY aantal\_klanten DESC  
LIMIT 10;



1. Geef de top 10 klanten volgens hun betalingen in de maand juli 2005. Kolommen zoals hieronder, sorteer op ‘betaald’, aflopend.

select customer.customer\_id, customer.first\_name,  
customer.last\_name, sum(amount) as betaald  
from customer  
 INNER JOIN payment ON customer.customer\_id = payment.customer\_id  
WHERE MONTH(payment\_date) = 7 AND YEAR(payment\_date) = 2005  
GROUP BY customer.customer\_id  
ORDER BY betaald DESC  
LIMIT 10

